

## **Appendix A**

```
// File: gbLogicAPI.h

#ifndef _GBLOGICAPI_H
#define _GBLOGICAPI_H

/*
 * Data types
 */

// return status for all GB watermarking logic calls
typedef enum {
    GB_nil,        // unexpected error condition
    GB_ok,         // success condition
    GB_deny,       // incoming segment to be dropped
    GB_prepare,    // cliient monitor is to perform synch
    GB_reauth,     // serv. mon. to request client resynch
} GB_action_t;

// 16-bit watermark data type
typedef struct { char bytes[ 2]; } GB_watermark_t;

// 256-bit shared secret state agreement data type
typedef struct { char bytes[ 32]; } GB_agreement_t;

// opaque context for watermarking logic module
typedef struct GB_context_s GB_context_t;

/*
 * Housekeeping
 */

// constructor
GB_context_t * gbInitialize( );

// destructor
void gbFinalize( GB_context_t * ctx);

/*
 * Client calls
 */

// return values: GB_nil, GB_ok
GB_action_t
gbPrepareWMark( GB_context_t * ctx,
                GB_agreement_t * data,
                ipaddr_t src,
                ipaddr_t dst);
```

```

// return values: GB_nil, GB_ok, GB_prepare
GB_action_t
gbWMarkOutgoing( GB_context_t * ctx,
                  GB_watermark_t * mark,
                  ipaddr_t src,
                  ipaddr_t dst);

/*
 * Server calls
 */

// return values: GB_nil, GB_ok, GB_deny
GB_action_t
gbSynchronizeWMark( GB_context_t * ctx,
                    GB_agreement_t const * data,
                    ipaddr_t src,
                    ipaddr_t dst);

// return values: GB_nil, GB_ok, GB_deny, GB_reauth
GB_action_t
gbWMarkIncoming( GB_context_t * ctx,
                  GB_watermark_t const * mark,
                  ipaddr_t src,
                  ipaddr_t dst,
                  void const * segment_hdr);

#endif /* _GBLOGICAPI_H */

// End of file

```